CLAIMS

WHAT IS CLAIMED IS:

1. A method to permit an application developer to decide the address space model to be supported by an operating system on a computer, the method comprising:

selecting one of a mostly private address space (MPAS) model and a mostly global address space (MGAS) model, where if the MPAS model is selected, then a process is permitted to map a shared object in a mostly private address space (MPAS) layout so that the process perceives a behavior as if the process is running on a multiple address space operating system, and where if the MGAS model is selected, then the process is permitted to map a shared object in a mostly global address space (MGAS) layout so that the process perceives a behavior as if the process is running on a single address space operating system.

2. The method of claim 1, wherein the selected MPAS model or MGAS model is indicated in a binary in an application.

3. The method of claim 2, wherein a kernel of the operating system reads the binary to determine if support will be provided for the MPAS model or the MGAS model.

4. The method of claim 1, wherein the MPAS layout comprises one of a 32 bit MPAS layout and a 64 bit MPAS layout.

5. The method of claim 4, wherein the 32 bit MPAS layout comprises private address spaces.

6. The method of claim 4, wherein the 64 bit MPAS layout comprises private address spaces and a shared address space.

7. The method of claim 1, wherein the MGAS layout comprises one of a 32 bit MGAS layout and a 64 bit MGAS layout.

8. The method of claim 7, wherein the 32 bit MGAS layout comprises shared address spaces and a private address space.

9. The method of claim 7, wherein the 64 bit MGAS layout comprises shared address spaces and private address spaces.

10. The method of claim 1, wherein the MPAS layout and the MGAS layout is partitioned in memory by a kernel of the operating system.

11. The method of claim 1, further comprising:

managing a hash page table, where a least recently used (LRU) algorithm is used to remove inactive or least recently used translations from the hash page table.

12. The method of claim 1, further comprising:

allocating a virtual address space for a 32 bit MPAS process by a method comprising:

mapping, by a process, an object to a first virtual address that allows efficient large pages to be stored; and

if the first virtual address is not available, then mapping the object to any virtual address.

13. The method of claim 1, further comprising:

(a) if a first process has mapped an object in virtual memory, then allocating a same virtual address in an address space of a second process, where the same virtual address is in an address space of the first process;

(b) if step (a) is not possible, then choosing a second virtual address that is aligned with (vaddr) which

is the virtual address for the object for the first process;

(c) if step (b) is not possible, then choosing a third virtual address which is aligned with vaddr such that large pages can be chosen;

(d) if step (c) is not possible, then choosing a fourth virtual address which is aligned with vaddr such that efficient aliasing is permitted; and

(e) if step (d) is not possible, then choosing any suitable virtual address.

14. The method of claim 1, further comprising:

if a plurality of processes have mapped an object in virtual memory, then allocating any suitable existing virtual address to align.

15. A computer system to permit an application developer to decide the address space model to be supported by an operating system on a computer, the computer comprising: comprising:

a processor; and

an operating system that can be executed by the processor, where the operating system is configured to select one of a mostly private address space (MPAS) model

and a mostly global address space (MGAS) model, where if the MPAS model is selected, then a process is permitted to map a shared object in a mostly private address space (MPAS) layout so that the process perceives a behavior as if the process is running on a multiple address space operating system, and where if the MGAS model is selected, then the process is permitted to map a shared object in a mostly global address space (MGAS) layout so that the process perceives a behavior as if the process is running on a single address space operating system.

16. The apparatus of claim 15, wherein the selected MPAS model or MGAS model is indicated in a binary in an application.

17. The apparatus of claim 16, wherein a kernel of the operating system reads the binary to determine if support will be provided for the MPAS model or the MGAS model.

18. The apparatus of claim 15, wherein the MPAS layout comprises one of a 32 bit MPAS layout and a 64 bit MPAS layout.

19. The apparatus of claim 18, wherein the 32 bit MPAS layout comprises private address spaces.

20. The apparatus of claim 18, wherein the 64 bit MPAS layout comprises private address spaces and a shared address space.

21. The apparatus of claim 15, wherein the MGAS layout comprises one of a 32 bit MGAS layout and a 64 bit MGAS layout.

22. The apparatus of claim 21, wherein the 32 bit MGAS layout comprises shared address spaces and a private address space.

23. The apparatus of claim 21, wherein the 64 bit MGAS layout comprises shared address spaces and private address spaces.

24. The apparatus of claim 15, wherein the MPAS layout and the MGAS layout is partitioned in memory by a kernel of the operating system.

25. An apparatus to permit an application developer to decide the address space model to be supported by an operating system on a computer, the apparatus comprising:

means for permitting a process to map a shared object in a mostly private address space (MPAS) layout so that the process perceives a behavior as if the process is running on a multiple address space operating system if a MPAS model, and for permitting the process to map a shared object in a mostly global address space (MGAS) layout so that the process perceives a behavior as if the process is running on a single address space operating system if a MGAS model is selected.

26. An article of manufacture, comprising:

a machine-readable medium having stored thereon instructions to:

permit a process to map a shared object in a mostly private address space (MPAS) layout so that the process perceives a behavior as if the process is running on a multiple address space operating system if a MPAS model, and to permit the process to map a shared object in a mostly global address space (MGAS) layout so that the process perceives a behavior as if the process is running

on a single address space operating system if a MGAS model is selected.